W.R.I.S.T.

(Weally, Really Incredible Spring capsTone project)

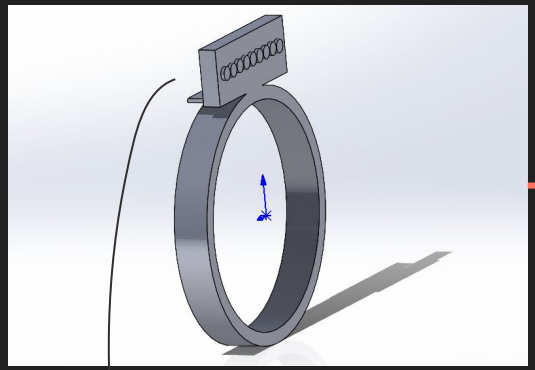
Final Presentation

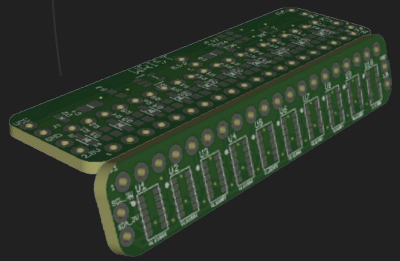
Team D5

Use Case and Requirements

- Want to create a **new input device** to make manipulating 3D models a more immersive experience
  - Key Insights:
    - Everyone knows how to use a trackpad
    - Everyone owns a 2D display
- Enable **mobility** in scenarios where a **3D visual aid** is present
  - i.e. engineering, architecture, chemistry demonstrations

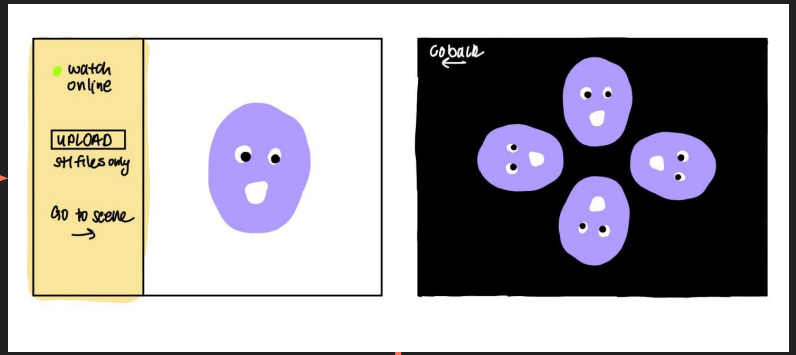
Solution Approach

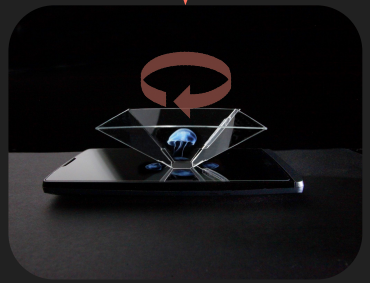




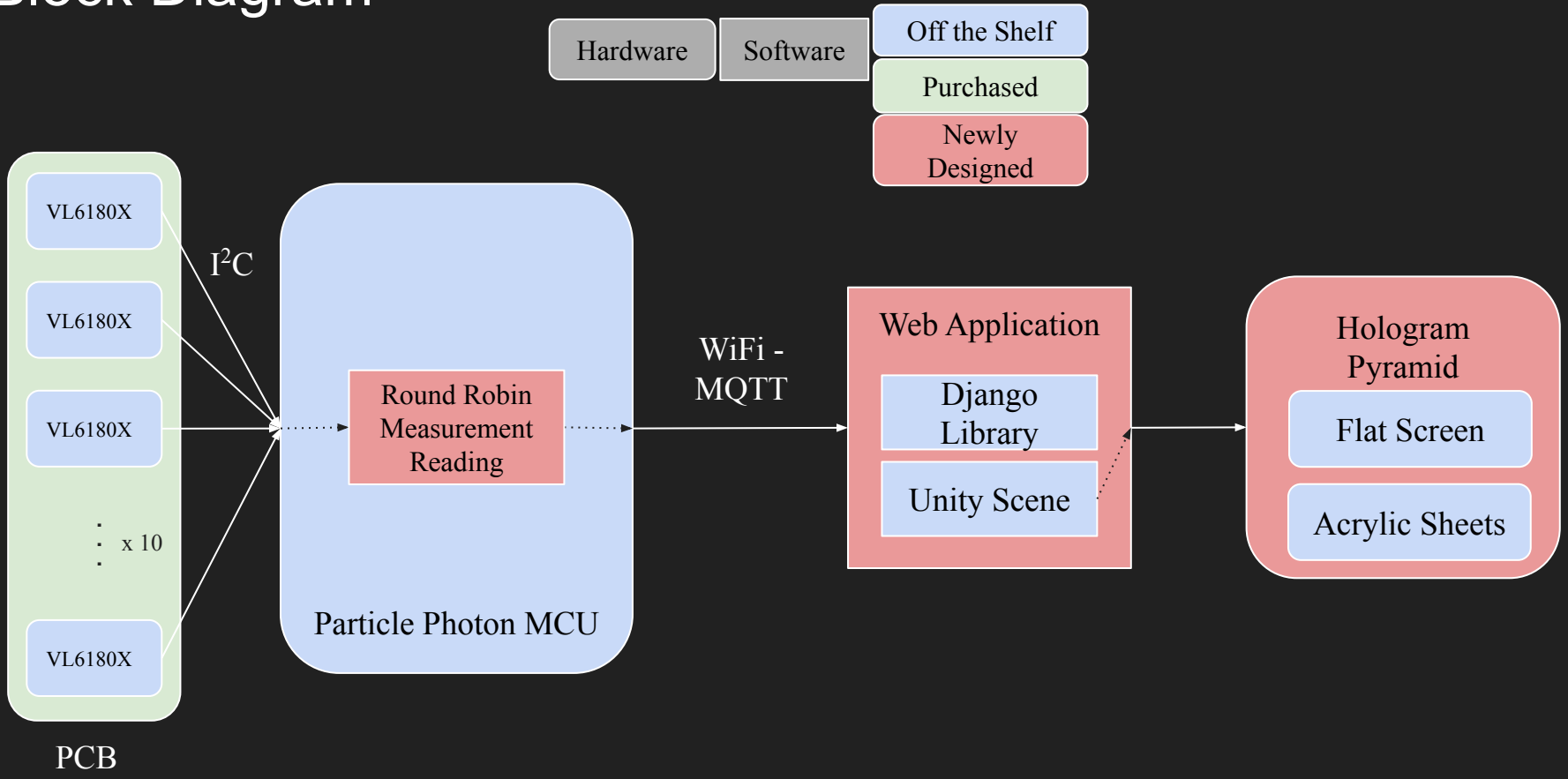
Possible Gestures:

- One finger swipe left/right/up/down
- Two finger swipe left/right



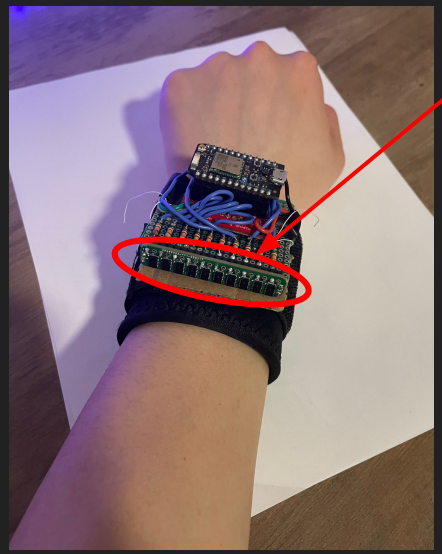


Block Diagram

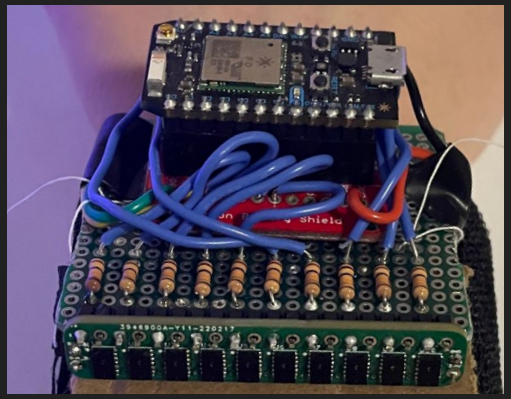


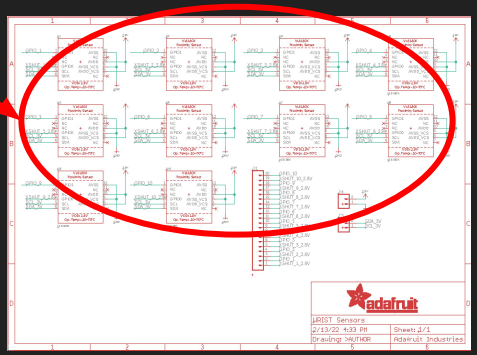
Complete Solution - Hardware

- **Build PCB off of this design**
- **Originally planned to make two PCB's**
  - Sensor array + "support components"
  - Ended up not really needing the "support components"
- Particle Photon MCU for WiFi



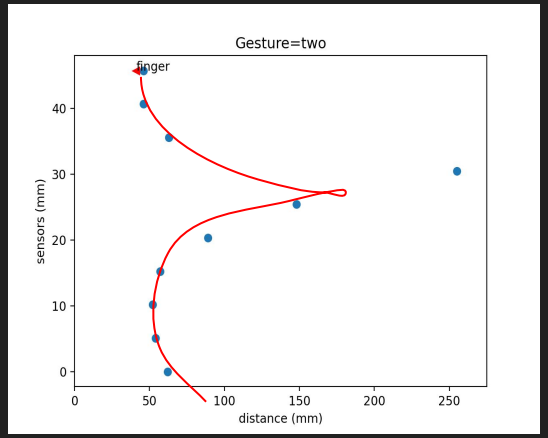
10 VL6180X Distance Sensors



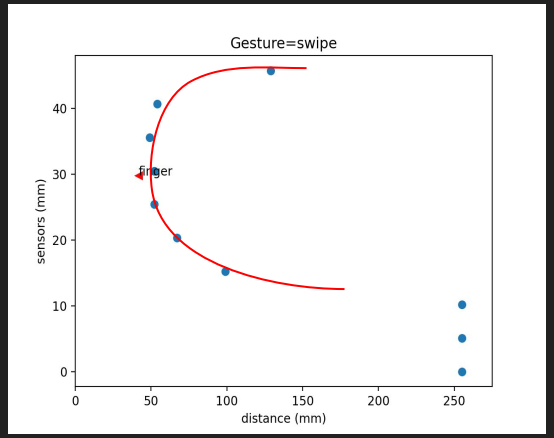


Schematic

Complete Solution - Finger Detection Using SVM



SVM learns that two fingers have a “w” shape



SVM learns that one finger swipes have a “u” shape

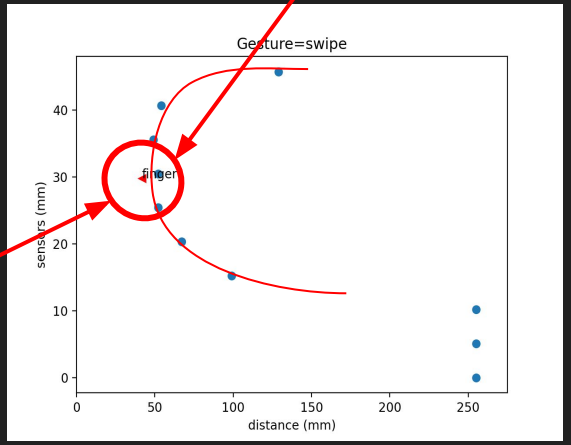
To find x:

Fit parabola to one finger data and take minimum x value

To find y, take a “weighted average”:

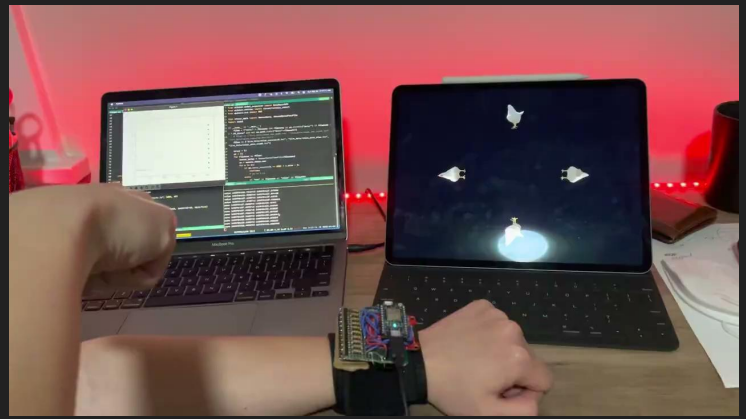
w_i = 1 / (x - x_i + 3)

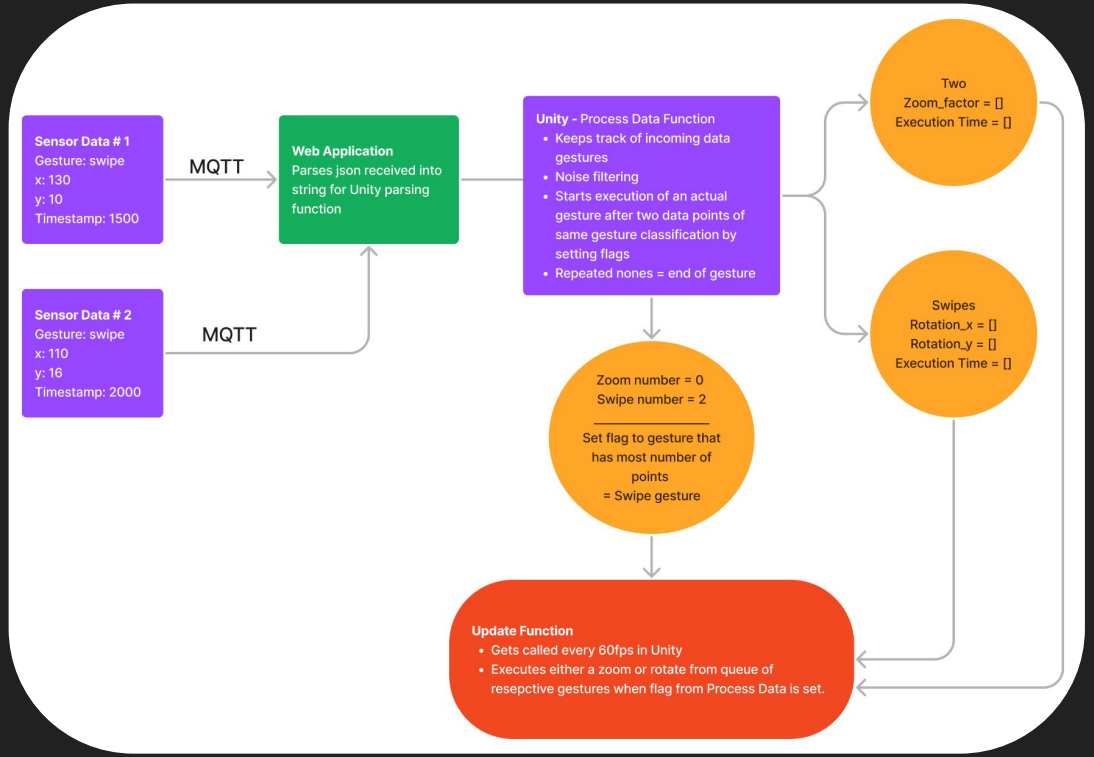
y = sum(x_i * w_i) / sum(w_i)



Complete Solution - Model Translation

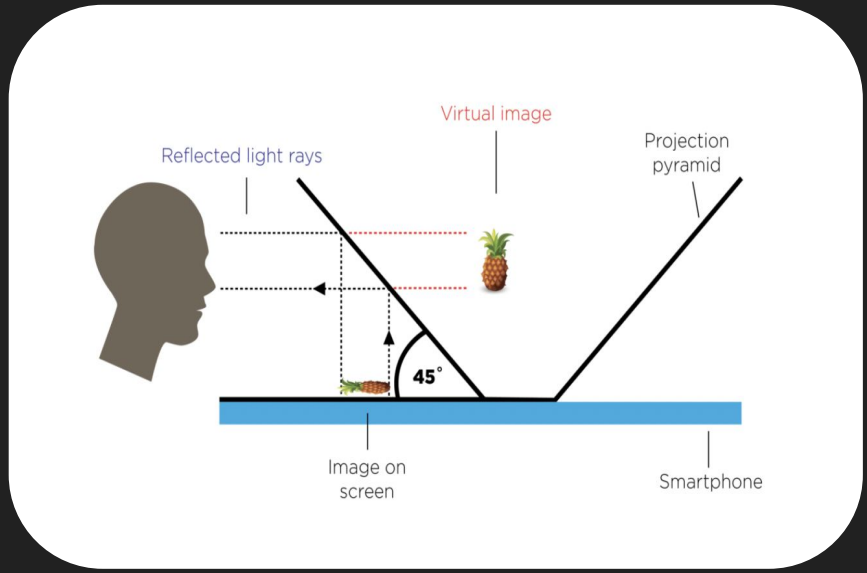
- Sensor data sent via MQTT to Web Application which calls Unity functions to process the data
- Noise filtering done as data gets sent into Unity
- Queues for respective gesture that detail rotation/zoom factor and execution time.

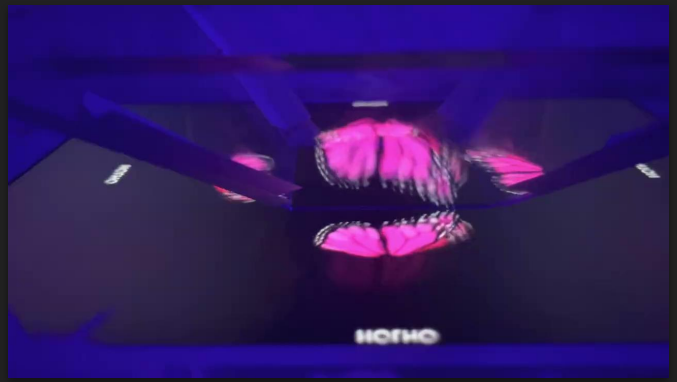


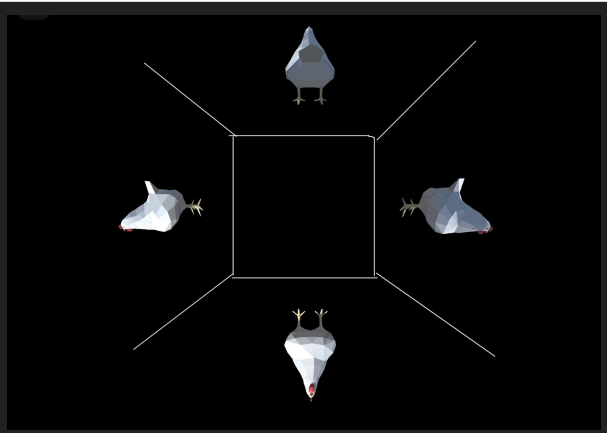


Complete Solution - Hologram

- Project four views of a 3D object onto 4 sides of the Pyramid.
- Will give an effect like the person is viewing the image in 3D.
- Need dark environment, so we created an encasing around display.



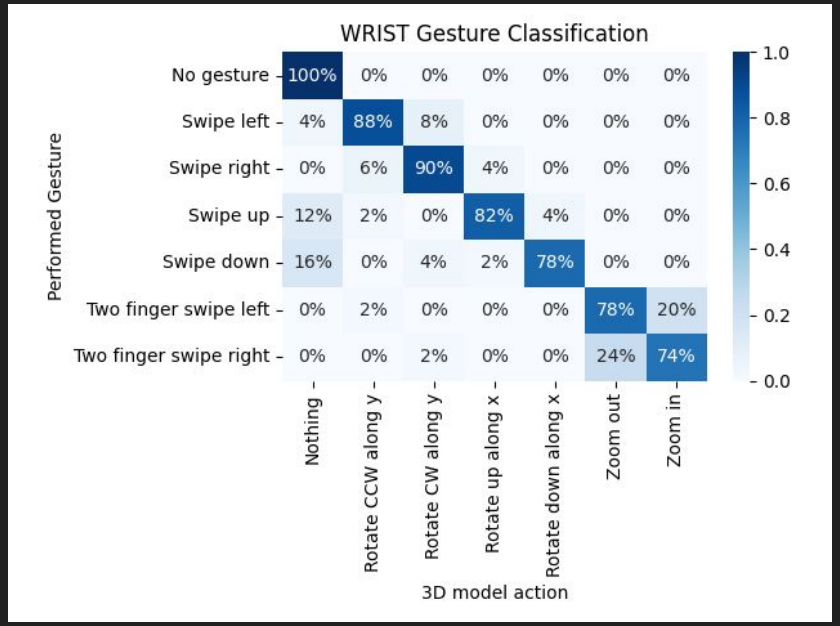




Tests

REQUIREMENT	TEST	TARGET METRIC
Accuracy of Gesture Detection	Have user swipe with one and two fingers and see if the proper gesture was detected	Zooming/pinching: 75% Rotation/swiping: 90%
Correctness of Gesture Detection	Measure actual displacement of swipes and see if measured displacement matches actual displacement	< 15% error
Latency	Timestamp all incoming and outgoing data and compare elapsed time in software	< 100 ms

Gesture Detection and Finger Tracking Results



Gesture-Action Confusion Matrix.

Note: 50 trials per gesture.

GESTURE	DISTANCE ERROR (%)
Swiping across sensors	20.29
Swiping along sensors	19.40
Two finger swiping across sensors	13.42

Finger tracking distance error table.

Note: Error was calculated using (measured - actual) / actual * 100

Latency

Time to collect all sensor datas: **~40ms**

Wearable → Middleware: **~29ms**

Middleware → WebApp: **~34ms**

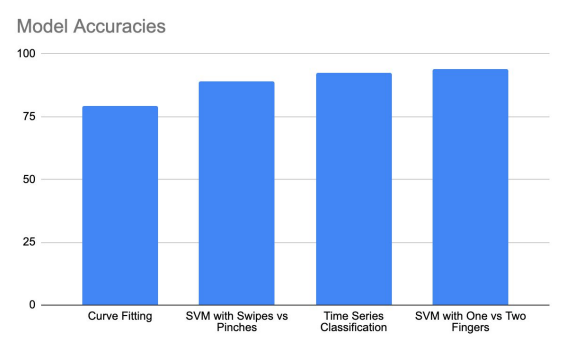
Total Latency: **~103ms**, about a **10Hz** update rate

Collected by averaging 100 timestamp differences during normal operation:

Note: during periods of “nones”, latency increases, because sensor data collection is slower. These tests are done during long periods of swipes.

Accuracy and Correctness Tradeoffs Through Gesture Algorithm

Model	Purpose	Shortcoming
Curve Fitting	Identifies gesture by curve fitting within a certain margin of error	Low accuracy - There is too much noise
SVM with Swipes vs Pinches	Identities gestures using machine learning from data of swipes and pinches	Low correctness - Pinch out looks like swipes towards the end of the gesture
Time Series Classification	Identifies gestures using machine learning from the a series of data	Low accuracy - The window of time is 24 points, but the average is 10 points
SVM with One vs Two Fingers	Identifies how many fingers are present from data of one and two finger swipes	High accuracy and correctness



Project Management

- 6 main tasks
  - Edward: Board Creation & Communication
  - Joanne: Unity & Hologram
  - Anushka: Gesture Recognition & WRIST Band Prototyping
  - LOTS of overlap
- Remaining things include refining and testing gesture detection algorithm and prototyping the final hologram

Task Name	Status	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14
Pre-Build	Complete												
Phase 1: PCB	Complete												
Phase 2: Unity	Complete												
Phase 3: Hologram	Complete												
Phase 4: WRIST Band	Complete												
Phase 5: Communication	Complete												
Phase 6: Gesture Algorithm	In Progress												
Phase 7: Testing	In Progress												
Final Check	In Progress												

Condensed Version of Schedule